Annotating Image ROIs with Text Descriptions for Multimodal Biomedical Document Retrieval

Daekeun You, Matthew Simpson, Sameer Antani, Dina Demner-Fushman, George R. Thoma

National Library of Medicine, National Institutes of Health, Bethesda, MD 20894

ABSTRACT

Regions of interest (ROIs) that are pointed to by overlaid markers (arrows, asterisks, etc.) in biomedical images are expected to contain more important and relevant information than other regions for biomedical article indexing and retrieval. We have developed several algorithms that localize and extract the ROIs by recognizing markers on images. Cropped ROIs then need to be annotated with contents describing them best. In most cases accurate textual descriptions of the ROIs can be found from figure captions, and these need to be combined with image ROIs for annotation. The annotated ROIs can then be used to, for example, train classifiers that separate ROIs into known categories (medical concepts), or to build visual ontologies, for indexing and retrieval of biomedical articles.

We propose an algorithm that pairs visual and textual ROIs that are extracted from images and figure captions, respectively. This algorithm based on dynamic time warping (DTW) clusters recognized pointers into groups, each of which contains pointers with identical visual properties (shape, size, color, etc.). Then a rule-based matching algorithm finds the best matching group for each textual ROI mention. Our method yields a precision and recall of 96% and 79%, respectively, when ground truth textual ROI data is used.

Keywords: Biomedical image analysis, biomedical article retrieval, content-based image retrieval, image overlay recognition, figure caption analysis

1. INTRODUCTION

Conventional approaches for biomedical journal article retrieval have been text-based with little attention devoted to the use of images in the articles. Text-based retrieval can fairly retrieve relevant articles using only text data such as citations, figure captions, and/or discussion; however, text information is sometimes insufficient in determining the usefulness of a publication for clinical decision support (CDS) or education, while images often convey essential information for the decision. For this reason content-based image retrieval (CBIR)-based approaches have been getting significant research attention. CBIR technology, however, is yet to be adopted into widespread use, either commercially or within research or academic institutions, mainly due to the "semantic gap" (the gap between low-level visual features and high-level human interpretations of image semantics) [1].

Recently we have focused on hybrid (text and image) approaches and use of specific local image regions for CBIR instead of entire images. Authors frequently highlight specific image regions of interest (ROIs) by overlaid markers (pointers, letters, etc.) and mention the presence of the markers and medical concepts within the ROIs in figure captions and/or discussion text. We expect that these ROIs contain more important and relevant information than other regions in images for indexing and retrieval of biomedical images and articles. Figure 1 shows an image that has an arrow pointing to *a 10-mm nodular area of ground-glass attenuation* in a CT (Computed Tomography) scan [2]. One may expect that the local image region that is brighter than its neighborhood may be the most important region within the image.

We have developed several essential algorithms toward implementing the multimodal retrieval approach such as pointer recognition and ROI localization [3–5]. In this article we present another essential algorithm that combines image ROIs with their textual mention. In the example in Figure 1, our pointer recognition and ROI localization algorithms identified the arrow and ROI (red rectangle); however, the extracted ROI is not annotated yet with appropriate content that describes it and which could be extracted from the figure caption (i.e., *a 10-mm nodular area of ground-glass attenuation*). The algorithm that extracts textual information about the visual ROI such as type of pointer (*arrow*) and description (*a 10-mm nodular area of ground-glass attenuation*) from the figure caption is a separate process based on text processing techniques [6]. Our proposed method pairs visual

ROIs and textual ROIs (obtained separately) to annotate cropped ROIs with extracted text descriptions. By *visual* and *textual* pointers or ROIs, we denote pointers and local image regions or textual mentions (pointer type, size, color, etc.) that are identified by our algorithms that are based on image processing and text processing, respectively.

The remainder of this article is organized as follows. Section 2 provides a summary of our previous image processing-based algorithms and section 3 describes our pairing algorithm. Evaluation results and discussion appear in section 4, and conclusions and future work are given in section 5.

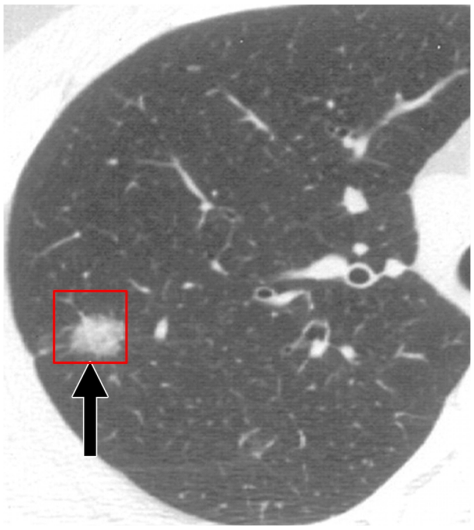


Figure caption: Transaxial thin-section (1-mm collimation) CT scan (lung window) obtained at the level of the apical segmental bronchus of the right upper lobe shows a 10-mm nodular area of ground-glass attenuation (**arrow**).

Figure 1. A CT scan image with an arrow pointing to the ROI

2. RELATED WORK

In our earlier publications on multimodal (text and image) biomedical retrieval systems [3–5], we discussed i) overlaid pointer detection and recognition, ii) local image regions of interest (ROIs) extraction, iii) a multimodal retrieval approach that utilizes visual features from ROIs, iv) textual feature extraction, and v) our continuing efforts to improve our initial algorithms. Our initial use of ROIs and visual features (e.g., color, texture, etc.) extracted from ROIs has been limited to re-ranking retrieval results obtained by conventional text- or CBIR-based retrieval approaches [5].

Recently we reported our work on the integration of textual and visual information extracted from ROIs pointed to by markers from biomedical articles. In [7] we introduced a visual ontology for biomedical image retrieval that defines a set of visual entities and the relationships among them, and that maps their appearance to textual concepts. Textual concepts and corresponding local image regions (ROIs) were automatically extracted and paired to create a visual ontology. A classification method that automatically labels image regions with appropriate concepts based solely on their appearance was developed to demonstrate the usefulness of our approach. Currently our research focuses on chest CT scan images with several frequently found concepts (disease/pathology) such as ground-glass opacity, honeycombing, and tree-in-bud pattern. We are now expanding our research to more image modalities and concepts.

3. METHOD

In this section we describe our algorithm that pairs visual pointers with textual mentions extracted from figure captions. Pointer extraction results from both image and captions are separately obtained in advance, and then combined by the proposed algorithm. Further details on pointer recognition and textual information extraction are discussed in [4, 7]. Figure 2 illustrates the structure of the proposed method.

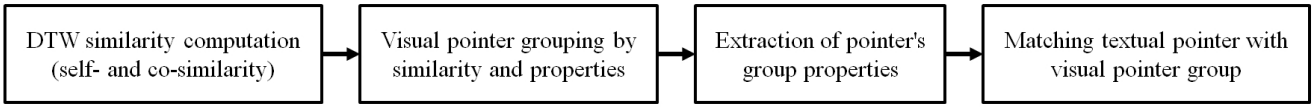


Figure 2. Our proposed pairing method

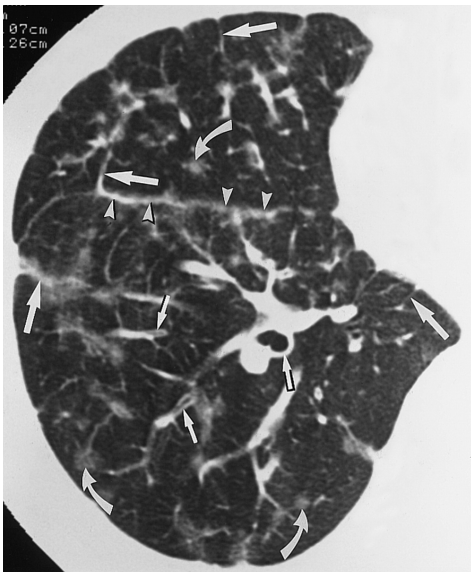


Figure caption: Hypereosinophilic syndrome in a 59-year-woman. Transverse thin-section CT scan (1-mm collimation) through the right lower lung demonstrates bronchial wall thickening (**small straight arrows**), interlobular septal thickening (**large straight arrows**), faint centrilobular nodules (**curved arrows**), patchy areas with ground-glass attenuation, and thickening of the interlobar fissure (**arrowheads**).

Figure 3. An example with four different groups of pointers (image appears in [8])

3.1 DTW Similarity computation

The first step in our method is to compute similarity scores in order to group recognized pointers into several groups in which pointers have identical properties. Figure 3 is a sample image that shows the necessity of pointer grouping before matching with textual pointers. The pointer recognition algorithm successfully detected 13 pointers (out of 14) and the figure caption mentions them as four different types of pointers, viz., *small straight arrows*, *large straight arrows*, *curved arrows*, and *arrowheads*. In simple cases, for example an image with only one black arrow, one white arrowhead, and one curved arrow, pointers may be easily paired with corresponding textual mentions by simply comparing pointer properties such as type and/or color. In the example shown in Figure 3, however, such a simple method may not provide satisfactory results. The image has three different types of pointers (straight arrow, arrowhead, and curved arrow) and arrows come in two different sizes (small and large). The arrowheads and curved arrows can be linked with corresponding textual concepts directly; however, straight arrows should be compared with each other to separate "small" from "large". Another important reason for pointer grouping is for noise filtering. Pointer recognition results may have some noisy pointers and they should not be paired with textual pointers. For example, in case the pointer recognition detects several noisy white arrowheads from the image in Figure 3, pairing without grouping may assign the text description (*interlobar fissure*) to the noisy arrowheads as well, which may result in irrelevant retrieval results. Grouping pointers could separate true pointers from noisy pointers (noisy pointers may be grouped into one or more groups as well). Even though grouping is performed, we still may need to find a true arrowhead group among the noisy groups; however, the chance of obtaining correct results can be increased by the grouping.

We apply dynamic time warping (DTW) [3] to obtain similarity scores between two different sequences of pointer boundary points. Since DTW finds an optimal alignment between two time-dependent sequences, it is necessary to rotate pointers so that they point in the same direction (e.g., upright) to obtain correct similarity matching scores. Figure 4 illustrates a rotated arrow and an arrowhead and the order of matching sequence (by dotted lines with arrow end). The (x,y) coordinates of the rotated contour points (solid lines) are consisting of the input sequences of the DTW matching.

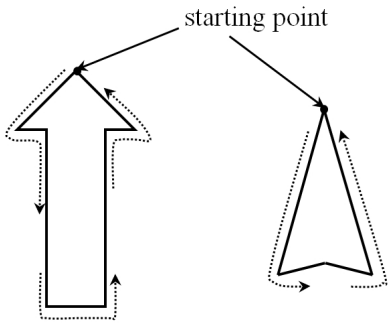
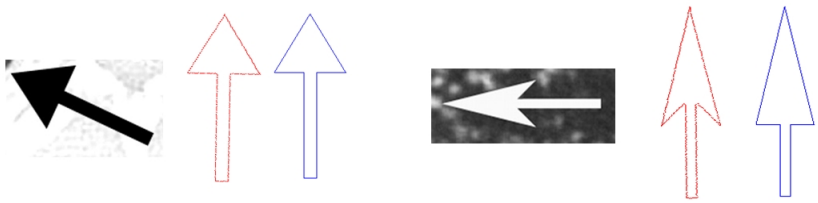
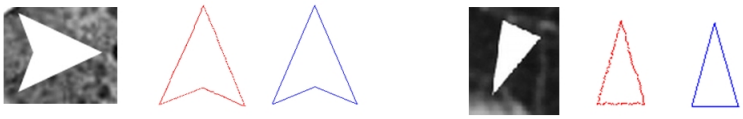


Figure 4. Illustration of rotated pointer boundary, start point, and sequence order for DTW matching

Two similarity scores are computed for each recognized pointer. One is a self-similarity score that measures the similarity between a pointer and its instant model pointer. An instant model pointer is one that is instantly created from the recognized pointer based on its pointer type and labels of line segments defined in [4]. In our algorithm only straight arrows and arrowheads are compared with their instant models since curved arrows have various tail shapes and hence it is difficult to create their instant models. Straight arrows and arrowheads have shape variation as well; however, these are minor and may be ignored. Figure 5 shows several sample instant models. Straight arrows have only one model shape and arrowheads have two different model shapes that differ only in the bottom part, as shown in the model in Figure 5(b). The length of the head and tail in straight arrow models are determined by the line segments that are labeled by the corresponding head or tail segment labels [4]. The DTW similarity score is obtained between a pointer and its instant model and is used as a self-similarity score. This score can be used not only for pointer grouping but also for noise removal. Pointers with larger similarity score (e.g., >0.2) can be eliminated as noise.



(a) Straight arrows



(b) Arrowheads

Figure 5. Instant model pointers (left: input, middle: rotated recognized pointer, right: instant model)

Another measure called the co-similarity score is useful for grouping pointers. Authors tend to use one or several identical pointers to highlight one or several ROIs in an image, but they can be annotated with a single medical concept. As shown in Figure 3, for example, all four large straight arrows are pointing to four different local regions, but they all indicate ROIs of one identical concept (*interlobular septal thickening*). Hence similarity in appearance may be the most useful feature in pointer grouping. Pointers with identical shape (pointer type and size) can then be distinguished by color (e.g., white or black), which occurs frequently. Co-similarity is measured by comparing two rotated pointer boundaries directly by DTW, and the distance measure is then

normalized by a product of their boundary length. Any two pointers with identical shape and size have a smaller similarity score, while two pointers of different size or shape are expected to have a larger score.

Besides the two similarity scores, each pointer is scored by its own visual features such as color and boundary length as well. In most cases pointers are monochrome and hence pointers whose color is closer to pure black or white (gray intensity 0 or 255) are more likely to be true pointers. Pointer boundary length is another good feature for noise removal. The score is proportional to the boundary length.

3.2 Pointer grouping

Pointers are grouped by a rule-based method based on the two similarity scores. In our method self-similarity is mainly used for noise removal before the grouping step. The Co-similarity score is used to determine whether a pointer can be added to a group or not. Figure 6 shows the pseudo code for our grouping algorithm.

In Figure 6 pointers i and j are two different recognized pointers. *self_sim*(i) and *co_sim*(i,j) denote self- and co-similarity of pointer i and pointer pair (i,j), respectively. *p_list* is a list where each entry has pointers (i,j) and *g_list* is a group list where each entry *g_list*[ ] has at least two or more pointers that are assumed to be identical. *g_list*[ ].*min_co_sim* is the minimum co-similarity among all member pointers in the group and T_1 and T_2 are thresholds. *diff* and *min* return the difference and minimum between the two input values, respectively.

Several group properties are computed as well from pointers in each group. They include averages of boundary length, tail width, pointer height, self-similarity, and co-similarity. Properties such as length, height and width are useful features that can be used to distinguish pointers and groups with size-related properties such as thin/thick, large/small, and long/short.

3.3 Pairing textual and visual pointers and ROIs

The purpose of our pairing algorithm is to find a visual pointer group identical or close to a textual pointer mention. The reality, however, is that noisy and/or missing pointers exist in both visual and textual pointers. Such errors can occur due to, for example, failure of our extraction algorithms or mistakes made in the publication.

Our approach is to search pointer groups and compute the matching score between a textual ROI and each group to find the best matching visual pointer group with the textual ROI. Figure 7 shows an example of a textual ROI. Each textual ROI has six fields, viz., "marker", "description", "shape", "color", "size", and "plural". The "marker", "shape", "color", and "size" together describe pointer appearance. For example, *large black straight arrow* has "size", "color", "shape", and "marker" properties, respectively. The "description" generally contains medical concepts seen in the ROIs (an example shown in Figure 7). The "plural" denotes whether multiple pointers in the image refer to the same textual description.

Visual pointers with specific properties such as size are considered first. For example, Figure 3 shows two groups of straight arrows, i.e., "large" and "small". Recognition results basically provide the contour length of each pointer, and additional size-related properties are computed as mentioned in section 3.2; however, they are not sufficient to distinguish large and small straight arrows. Size is a relative characteristic and all straight arrow pointer groups need to be compared to find pointers that are larger than others. In the grouping results for the sample image in Figure 3, there may exist only two straight arrow groups, which is an ideal and easier case, or more than two, where we need to consider more visual features to choose two true pointer groups. For textual pointers with the size field set, we first compute matching scores with each of the pointer groups and find the top two groups with the highest scores. Then we compare the size-related characteristics between the two and choose the proper one for the textual pointer (e.g., larger of the two for *large straight arrows*). When one straight arrow group is found, we may not be able to pair the group with the correct textual pointer since no clue is available for choosing *large* arrows against *small* arrows.

Textual pointers without relative pointer characteristics are then paired with the remaining pointer groups that are not paired. Such cases are much simpler, and scoring them by comparing their pointer type, color, and plural properties is sufficient to choose the best matching visual pointer group. Additional features such as average contour length and self- and co-similarity scores may contribute to improve the scores to find true pointer groups.

```
1. // pointer pair entry generation
   for each pointer i {
       for each pointer j {
           if self_sim(i) > 0.2 OR co_sim(i, j) > 0.3
                continue;
           entry_score(k) = self_sim(i)+co_sim(i,j);
           Add (i,j) to pointer list p_list[k];
           k++;
       }
   }

2. Sort the entry list from minimum to maximum by entry_score;

3. Add p_list[0] to group list g_list[0];

4. // pointer grouping
   for each entry in p_list starting from index 1 {
       if g_list[m] contains pointer i (or pointer j) {
           if diff(co_sim(i,j), g_list[m].min_co_sim) < T1 {
                add pointer j (or i) to g_list[m];
                g_list[m].min_co_sim = min(co_sim(i,j), g_list[m].min_co_sim);
           }
           continue;
       }

       if g_list[m] contains both pointer i and j
           continue;

       if both pointer i and j are not found in g_list[ ] {
           create a new pointer group g_list[g_num] and add pointer i and j to g_list[g_num];
           g_num++;
           continue;
       }

       if both pointer i and j are found in g_list[ ] but from two different groups {
           if diff(co_sim(i,j), min(g_list[k1].min_co_sim, g_list[k2].min_co_sim)) < T2
                merge two groups;
           else
                continue;
       }
   }
```

Figure 6. Pointer grouping algorithm

[ROI{marker='arrow', description='mild dilatation of bronchi', shape='', color='', size='', plural=true}]

Figure 7. An example of textual ROI representation

Curved arrows and asterisk symbols cannot be paired by the aforementioned method since self- and co-similarity are not available for them. Instead, basic visual properties such as type, color, and contour length

are used to group them. It is our observation that curved arrows and asterisk symbols are rarely used with size-related properties. Hence a method similar to the matching method used for pointers without relative characteristics is sufficient for finding a best matching textual ROI for them.

4. EVALUATION

4.1 Dataset and evaluation method

Our dataset contains 298 chest CT images found in ImageCLEF2010 [9] and used in [7]. All these images contain one or more pointers and a ground truth set containing visual and textual extraction results for each pointer was created in a semi-automatic way. Two ground truth files, one for visual ROIs and the other for textual ROIs, were created separately by our pointer recognition and text processing algorithms, respectivley. To create the visual ROI ground truth, we first automatically recognized pointers in the images by our pointer recognizer, and then manually examined the result to eliminate noisy pointers, add missing pointers, and amend incorrect recognition results. Then the visual ROI ground truth was combined with the textual ROI ground truth. Table 1 shows the number of each pointer type in our dataset. As shown in the table, about 96% of the pointers are straight arrows and arrowheads.

Table 1. Pointers in the dataset

	Straight arrow	Curved arrow	Arrowhead	Asterisk
Number	693	23	316	17
Total	1,049			

Two different textual ROI data, viz., *Actual* and *Ground truth*, were used in this evaluation. *Actual* data contains errors such as missing pointers or incorrect pointer properties. For example, every image has at least one pointer; however, 50 images in *Actual* data have no extracted textual ROIs. The *plural* field (see Figure 7) is frequently incorrect in *Actual* data as well. More details of text processing methods, results, and error analysis are discussed in [7].

A result image as shown in Figure 8 was created for every input image. Only pointers that are correctly coupled with textual ROIs are counted as success. We do not consider the description field in our evaluation because it is not needed for pairing pointers in an image with their textual mentions. The effectiveness of our textual extraction methods is addressed in our previous work [7] and is beyond the scope of this article.

4.2 Evaluation result

Table 2 and 3 show our evaluation results. Table 2 gives pointer recognition result that is obtained solely by our pointer recognition algorithm. Table 3, on the other hand, shows recognition results after textual ROIs are included. For each use of textual data, two evaluation methods are used to evaluate our pairing method. The *Individual* method counts pointers that are successfully recognized by combined visual-textual result; however, text ROI linked to the pointers are not examined and hence several of them may not be paired with correct textual ROIs. The *Paired* method includes those successfully recognized by visual-only recognition and paired by correct textual mentions through the pairing algorithm. Figure 8 is a good example to explain the difference. In the example three pointers, two arrows and one arrowhead, are recognized by visual-only recognition and four ROIs are extracted from figure caption (hence *3* are *Detected*). After textual ROI data combined with the visual detection through the pairing algorithm, all three pointers remain in the final result since both "arrow" and "arrowhead" are mentioned in the figure caption (and hence included in the textual ROIs). Hence *Individual* yields *3* in this example. However, there is an error in the pairing result. *ROI 0* in the text result indicates that it is pointed to by a "short arrow" which is not recognized, and the "open arrow" is coupled with *ROI 0* (see the pairing result overlaid on the image). Hence only two, the *ROI 1* and *ROI 2* pointed to by "long arrow" and "arrowhead", are correctly paired pointers and they are counted in the *Paired* method.

By comparing the numbers of detected pointers in Table 2 and 3, we notice some noise removal effect achieved by combining textual data with the visual-only recognition results. When a fairly accurate textual

Table 2. Visual-only pointer recognition result

	Ground truth	Total detected	Detected true	Precision (%)	Recall (%)
Number of pointers	1,049	1,017	856	84.2	81.6

Table 3. Visual and textual ROIs pairing algorithm evaluation result

Textual data	Evaluation method	Number of pointers (concepts)			Precision (%)	Recall (%)
		Ground truth	Detected	Detected true		
Actual	Individual	1,049	693	668	96.4	63.7
	Paired			636	91.8	60.6
Ground truth	Individual		856	830	97.0	79.1
	Paired			825	96.4	78.6

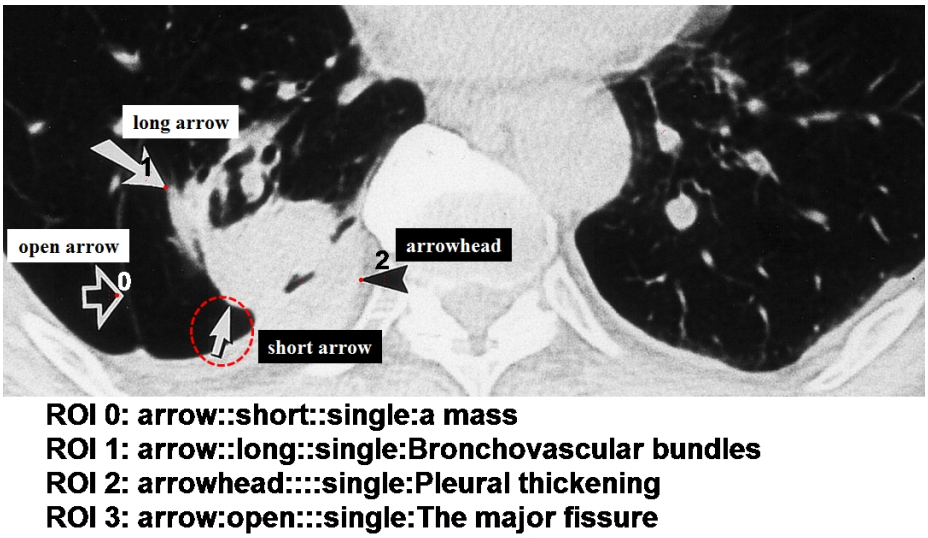


Figure 8. An example of pairing result. The short arrow for *ROI 0* (in the dotted circle) is not recognized and the open arrow for *ROI 3* is matched incorrectly to *ROI 0*. (original image appears in [10])

result is included, recognition precision is improved with some decrease in recall (see numbers in *Individual*). The performance variation is dependent on the accuracy of the combined textual data. As shown in Table 3, perfect textual results (*Ground truth*) increase the precision about 13% while recall is similar to the initial result with a slight decrease (81.6% to 79.1%). On the other hand, real textual data (*Actual*) achieves higher precision as well; however, it significantly decreases the recall compared to the visual-only recognition (81.6% to 63.7%).

These changes in precision and recall can be explained as follows. Assume that pointer recognition detects several straight arrows and arrowheads. Textual processing, however, extracts only "straight arrows" from the figure caption. In cases textual result is correct, those recognized arrowheads are noisy pointers and eliminating them increases the precision. Opposite cases, however, are the main cause of reduced recall in the combined recognition results. True arrowheads that are recognized from the image but not detected from the caption are eliminated, and this results in a decrease in recall. Undetected pointers in textual ROI result also may affect annotation process that utilizes the ROIs. In such cases it may be impossible to automatically annotate successfully extracted image ROIs with accurate descriptions that are most probably extracted from figure captions.

Figure 9 shows another error case due to undetected visual pointers. In the result *ROI 2*, which is indicated by a "thin arrow" shown in the dotted white circle, is coupled with the "thick arrow" in the text result since the "thin arrow" is not recognized. In such cases identifying size-related properties (e.g., thick/thin, long/short, etc.)

is impossible and hence the detected "thick arrow" may or may not be successfully paired with its corresponding textual result. Undetected pointers are responsible for all the missing five pointers in the result using *Ground truth* (in the *Detected true* column, 830 vs. 825).

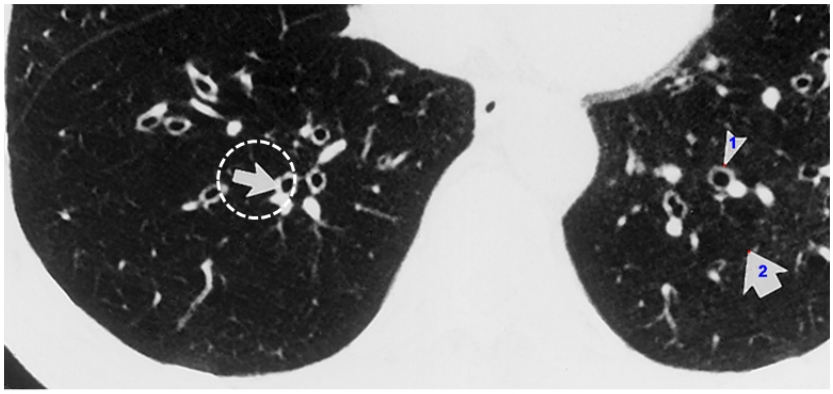
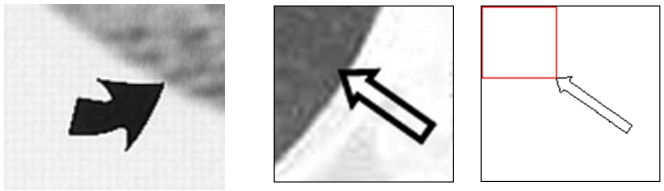


Figure 9. An error case due to an undetected pointer (original image appears in [11])



(a) An example of pointer type discrepancy

(b) Open arrow

Figure 10. Sample pointers causing pairing errors

Several pointer types were successfully recognized by visual-only recognition but not paired with corresponding textual ROIs. In ideal cases all successfully recognized pointers are expected to be coupled with textual ROI when the ground truth text ROI is used and the pointers are mentioned in the text. Besides the undetected pointer factor above, we identified two causes for this error: i) pointer type discrepancy between ground truth and extraction result, and ii) "open arrow" which is excluded in the pairing algorithm. Figure 10 shows sample pointers of the error cases. The arrow shown in Figure 10(a) is classified as a "straight arrow" in our ground truth. However, it is mentioned as a "curved arrow" in the figure caption. Authors frequently use the two names, i.e., "arrow" (sometimes with "straight") and "curved arrow", for the pointer and it is difficult to match one of the two text names to the visual arrow. Figure 10(b) shows an open arrow that is currently excluded in our pairing algorithm. Unlike general (solid) arrows (e.g., Figure 10(a)), open arrows have a solid (and thick) pointer contour, but the body region is not filled. In the example in Figure 10(b), our pointer recognizer successfully detected the boundary and recognized it as an arrow (with correct ROI location shown by a rectangle). In general, distinguishing open arrows from general (solid) arrows is a difficult problem.

Our pairing algorithm achieves high precision (the noise removal effect) in ROI extraction, which would lead to high precision in retrieval. As discussed above precision of image ROI extraction is significantly improved when textual ROI data is combined, either actual or ground truth. Also it achieves high precision in combining visual and textual ROIs. About 96% of successfully detected pointers are paired with correct textual mention. This is a promising result toward achieving high accuracy in annotating the ROIs and images containing them

with accurate text descriptions.

To enhance the performance of our initial pairing method, improvements from both pointer recognition and text processing need to be achieved. Our current system can achieve the highest recall of 81.6% in a case that all detected true pointers by visual-only recognition are paired with correct textual ROIs, which is almost impossible to achieve. Achieving high accuracy in textual ROI extraction and developing robust solutions to the identified error cases in the pairing algorithm need to be considered first. Improving recall rate of visual-only recognition, which would be rather difficult than the first two, would be the next task.

5. CONCLUSION

Local image regions in biomedical images may have more meaningful information and may be more relevant than other regions in an image for biomedical image and article retrieval. Authors frequently use pointers and symbols to highlight specific local regions and mention them in figure captions and text discussions. Detecting those pointers can help extract specific local regions of interest (ROIs), and using these ROIs could improve the relevance of conventional retrieval approaches by combining textual and image features from local regions.

Our prior efforts in ROI processing have been focused mainly on pointer recognition and ROI segmentation, which are purely image processing-based tasks. In this article we report our initial effort on combining visual and textual ROIs extracted from images and text data such as figure captions, respectively. Pairing visual ROIs with the corresponding textual mentions is the first step toward automatic indexing of the ROIs and images containing them. The tagged ROIs can then be used for image retrieval or building a visual ontology.

In this article we propose a DTW-based visual and textual ROI pairing algorithm. Our pairing algorithm combines visual pointers with textual mentions by grouping recognized pointers by their visual characteristics first, and then searching for the best-matching pointer group with a text mention. It successfully pairs over 96% of recognized true pointers with their textual mention when ground truth text data is used. To improve performance, both visual and textual pointer extraction need to be improved simultaneously. Improving the text ROI extraction algorithm, however, could be more powerful to improve the initial result. A successfully detected text ROI in an image could result in one or several pointers being successfully paired with their textual mentions.

ACKNOWLEDGMENTS

This research was supported by the Intramural Research Program of the Lister Hill National Center for Biomedical Communications, an R&D division of the National Library of Medicine, at the National Institutes of Health, U.S. Department of Health and Human Services. We would like to thank the ImageCLEF [9] organizers and the Radiological Society of North America (RSNA), publisher of Radiology and RadioGraphics, for making the database available for the experiments under the ImageCLEFmed medical image retrieval task.

References

[1] Deserno, T. M., Antani, S., and Long, R., "Ontology of gaps in content-based image retrieval," *Journal of Digital Imaging* **22**, 202–215 (April 2009).

[2] Lee, K. S., Jeong, Y. J., Han, J., Kim, B.-T., Kim, H., and Kwon, O. J., "T1 NonSmall Cell Lung Cancer: Imaging and Histopathologic Findings and Their Prognostic Implications," *Radiographics* **24**(6), 1617–1636 (2004).

[3] You, D., Apostolova, E., Antani, S., Demner-Fushman, D., and Thoma, G. R., "Figure content analysis for improved biomedical article retrieval," *Document Recognition and Retrieval XVI* **7247**, 72470V–10, SPIE, San Jose, CA, USA (2009).

[4] You, D., Antani, S., Demner-Fushman, D., Rahman, M. M., Govindaraju, V., and Thoma, G. R., "Biomedical article retrieval using multimodal features and image annotations in region-based CBIR," *Document Recognition and Retrieval XVII* **7534**(1), 75340V+, SPIE, San Jose, California, USA (2010).

[5] You, D., Antani, S., Demner-Fushman, D., Rahman, M. M., Govindaraju, V., and Thoma, G. R., "Automatic identification of ROI in figure images toward improving hybrid (text and image) biomedical document retrieval," *Document Recognition and Retrieval XVIII* **7874**(1), 78740K+, SPIE, San Francisco, California, USA (2011).

[6] Simpson, M. S. and Demner-Fushman, D., "Biomedical Text Mining: A Survey of Recent Progress," *Mining Text Data* , 465–517 (2012).

[7] Simpson, M. S., You, D., Rahman, M. M., Antani, S. K., Thoma, G. R., and Demner-Fushman, D., "Towards the creation of a visual ontology of biomedical imaging entities," *AMIA Annual Sympiosium Proceedings(accepted)* (2012).

[8] Johkoh, T., Mller, N. L., Akira, M., Ichikado, K., Suga, M., Ando, M., Yoshinaga, T., Kiyama, T., Mihara, N., Honda, O., Tomiyama, N., and Nakamura, H., "Eosinophilic Lung Diseases: Diagnostic Accuracy of Thin-Section CT in 111 Patients," *Radiology* **216**(3), 773–780 (2000).

[9] Müller, H., Kalpathy-Cramer, J., Eggel, I., Bedrick, S., Reisetter, J., Kahn, C. E., and Hersh, W. R., "Overview of the CLEF 2010 medical image retrieval track," *Working Notes of CLEF 2010* (2010).

[10] Partap, V. A., "The Comet Tail Sign," *Radiology* **213**(2), 553–554 (1999).

[11] Copley, S. J., Wells, A. U., Mller, N. L., Rubens, M. B., Hollings, N. P., Cleverley, J. R., Milne, D. G., and Hansell, D. M., "Thin-Section CT in Obstructive Pulmonary Disease: Discriminatory Value," *Radiology* **223**(3), 812–819 (2002).